UCRL-JC-133946
PREPRINT

Easy-to-Use Interfaces

M. M. Blattner
Y. Tong
D. O. Blattner

This paper was prepared for submittal to the
Institute for Electrical and Electronics Engineers
Symposium on Visual Language
Tokyo, Japan
September 13-16, 1998

April 1, 1999

Lawrence
Livermore
National
Laboratory

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

Easy-to-Use Interfaces

Meera M. Blattner
University of California, Davis,
Lawrence Livermore National Laboratory
Livermore, CA
blattner@llnl.gov

Youtang Tong
Dalian University of Technology
Dalian, China
jinsui@pub.dl.lnpta.net.cn

Douglas O. Blattner
Community Vision, Inc.
Las Vegas, NV
Doug.Blattner@CommunityVision.com

Abstract

Easy-to-use interfaces are a class of interfaces that fall between public access interfaces and graphical user interfaces in usability and cognitive difficulty. We describe characteristics of easy-to-use interfaces by the properties of four dimensions: selection, navigation, direct manipulation, and contextual metaphors. Another constraint we introduced was to include as little text as possible, and what text we have will be in at least four languages. Formative evaluations were conducted to identify and isolate these characteristics. Our application is a visual interface for a home automation system intended for a diverse set of users. The design will be expanded to accommodate the visually disabled in the near future.

1. Introduction

One of the problems driving research in user interfaces is that of creating interfaces which can be used by the general public without training, known as *public access systems.* Examples of such interfaces are found in public facilities such as airports, shopping malls, and museums. Generally interfaces that require a small amount of training to use them are found on computerized order forms, web browsers, children's interfaces, and computer games. We became interested in interfaces that require little or no training while designing interfaces for home automation systems.

Home automation interfaces are operated by every member of the household. The users may be very young or very old, disabled, non-English speaking, or those who have never used a computer. The similarity between public access interfaces and home automation interfaces is their diverse set of users. The dissimilarity is in that home automation interfaces are not in a public place, but in a home and some instruction may be introduced through a short tutorial (usually on video tape). Even though the characteristics of the residents may be known, the introduction of a new user, such as a visitor, can require a home automation system to be learned rapidly without training by someone with unknown characteristics. As computer usage becomes as common as the telephone or television and while applications grow more complex than those seen in public access systems, this type of interface, will become increasingly important. We call these *easy-to-use* (ETU) interfaces.

The present home automation technology is the management of information received from controllers using sensor-driven devices. (We are not using active badges or recognition with camera input.) Sensors can check light, temperature, moisture, sound, weight, pollutants, the presence of motion, location of objects, and, more generally, detectable changes in the status of the environment. The sensor-acquired information is sent to a control device that can alter the status (as with internal home temperature) or simply inform the user of the conditions (as with the external temperature).

It should be clear that to accomplish the tasks of controlling environment described above the interface must be complex with many choices and detailed specifications not required by publicly situated interfaces. Presently, most home automation interfaces are difficult to use and require not only a detailed knowledge of the functions, but some programming skills. One exception to this is the home automation interface designed by Plaisant and Shneiderman [11], who used touch screen design with direct manipulation and a controller that could be set by sliders, clocks and calendars. It was our goal to further simplify this interface even more to make home automation interfaces accessible to those who may not speak English and even less experienced in computer use than those using the Plaisant and Shneiderman interface.

2. Overview

In this project, we worked closely with Community Vision, a Las Vegas based home automation and community network company (http://www.CommunityVision.com). By the time we entered the project, a decision had already been made that a Micron touch-screen computer would be used for the central home automation interface. Although the authors are concerned about visually disabled users, in our prototype we chose to create an interface that would not be suitable for the visually disabled at this time. A design is in progress which will provide an auditory interface for the visually disabled. The focal point of our research is to examine the problems of designing an interface that required as little text as possible and still could be used by untrained users. The design was directed towards low cost home automation systems to be installed in new homes. There are only three functions available in this prototype: lighting, climate control, and security. Of these, the lighting function was the most complex and will be the only one described in this article. The home automation system has controls located in each room as well, however, this aspect will not be described here.

Our attempt to simplify traditional graphical user interface design and create an ETU interface was centered on these issues:

a. *Selection:* When selecting objects and functions, where and what do the users touch? Do the users understand the relationship between what they touch and the functions they desire?

b. *Navigation:* What navigational aids can be introduced to aid the user in making successive choices? How do the users understand the process of moving from one operation to the next as well as the refinement of a query?

c. *Dragging:* When direct manipulation is required, can a visual symbol be used to indicate how and when to drag objects? Do users understand the purpose of direct manipulation?

d. *Contextual Metaphors:* Can metaphors be introduced from devices we already use in the world around us. Are they are self-explanatory?

e. *Help:* Are help routines the same for ETU as for other interfaces? Are the help routines we use informative or confusing? What kind of help routines can be designed for ETU interfaces?

f. *Text:* Text should be reduced to a minimum so those who don't speak English or can't read can use the interface. How little text can be incorporated into the interface and still be self-explanatory? What icons can replace text?

3. Related work

3.1 Public access systems

In his book on public access systems, Kearsley [5] lists a number of user interface design guidelines. Among these are: provide a model or metaphor, be consistent, minimize navigation, users should always have control, and user responses should be acknowledged. Another set of guidelines are: state instructions clearly, provide default options, provide redundant response modes, allow people to change or confirm their responses, provide helpful feedback, and provide a help function. These guidelines are as critical to ETU interfaces as public access interfaces.

In our design however, we observed that it is not possible to make our interactions as obvious as in a public access system. For example, an ATM displays choices and expects the user to push a button. There is only one choice the user can make. Entering monetary amounts for deposit or withdrawal is done with a keypad, as is entering a pin number. An ATM is highly dependent upon the user reading text to make selections.

3.2 Children's Interfaces

Interfaces for children's games and educational programs are the closest to ETU interfaces we now have. Navigation through the program must be fairly self-evident. Navigation is either by arrows indicating backwards or forwards, or clicking on an object on the screen. An exit icon is usually present. Text must be reduced to a minimum unless the program is designed to teach reading. Voice output is often used for explanations of various kinds. Because children's software is found on standard computer hardware, children learn to use a mouse. Some experimental work is being done with providing educational software with pen interfaces [6]. Nintendo has a special input device with buttons that can be held with two hands. Children have short attention spans; bright colors and music keep them occupied.

3.3 Controls for an interactive game

Johnson [3] describes a problem that developers of an interactive game had with their controllers: they were non-intuitive, unsystematic, and too numerous. Johnson was hired as a consultant to design a new control system. The game's producers wanted to minimize or eliminate the use of text for their controllers. Johnson and an artist drew animated mime-figures to convey meaning. Johnson's problem was harder than ours, but the problems described by Johnson were pertinent to the ones we experienced, because we also tried to eliminate the use of text with animated icons and mime. Two of Johnson's conclusions are: (1) "A picture may be worth a thousand words, but finding the right picture to convey a verbal concept can be very hard. The best way to convey some verbal concepts is verbally." (2) "If symbols in a set depict their meanings well, people can discriminate and recognize them even if the set is large. Users may not even realize that the set is large." Our investigation into ETU interfaces confirms these two points.

3.4 Web pages and consumer products

Use of the web requires some computer experience, even though it may be minimal. Input to the web through a browser is with a mouse-controlled cursor, although there may be specialized input-output devices for those with special needs, such as screen readers used by those either with visual disabilities or other problems that prevent them from reading a screen. The web is still used primarily as an information resource, but retail sales and services are increasingly more important. Typed input is used in forms and pop-up menus are common. Web navigation is primarily through links. The clarity of the input instructions vary considerably, but it is easy to design pages that are self explanatory if the functionality is also simple. Web interfaces for various consumer products, digital libraries, and other applications, will put more demands on them, most of which will too complex to be considered ETU.

Consumer products, such as VCRs, have notoriously poor interfaces. Generally the interface can only be used after carefully reading an instruction booklet. If the booklet is not available, it is difficult to use the product except for its simplest functions.

4. Evaluation methods

A series of formative evaluations have been used to assist us in the development of our home automation interface. Formative evaluations differ from summative evaluations in that they are used to compare established usability specifications and produce qualitative data that can help determine what changes can be made to improve usability. The methodology and design of the formative evaluations used were similar to those in *Developing User Interfaces* [2].

Six subjects were evaluated. They were asked to perform nine particular tasks required for home automation functions. Four of the subjects were videotaped (one requested that she not be videotaped) and one subject, a human factors expert, was led through the process while discussing the various aspects and choices that were made. Five of the subjects were timed with a stopwatch. The prototype evaluated was designed on a Micron personal computer with keyboard and mouse. The cursor was to simulate the touch screen until the software could be transferred to a Micron touch screen computer. Two of the subjects had never used a computer and the coordination of the mouse proved to be an unnecessary complexity. The mouse movement was replaced by the subject touching the screen and the evaluator sitting next to the subject then moving the cursor to the desired location.

Four of the subjects were familiar with computer use. Two of these were office staff, one was a high school student, and one a usability expert. A nonuser is defined as one who has not used a traditional graphical user interface; kiosks, ATMs, and other public access systems are not included as computer use. One of the nonusers was a custodian and the other was a woman in her late seventies with a college degree. None of the subjects had disabilities. Those with even a limited amount of computer experience reacted very differently from those with no computer experience as described below. As grades K-12 introduce computers to children, we may expect to find very few nonusers. We can only speculate on the design implications of universal use of computers for ETU interfaces with computer use as widespread as textbooks.

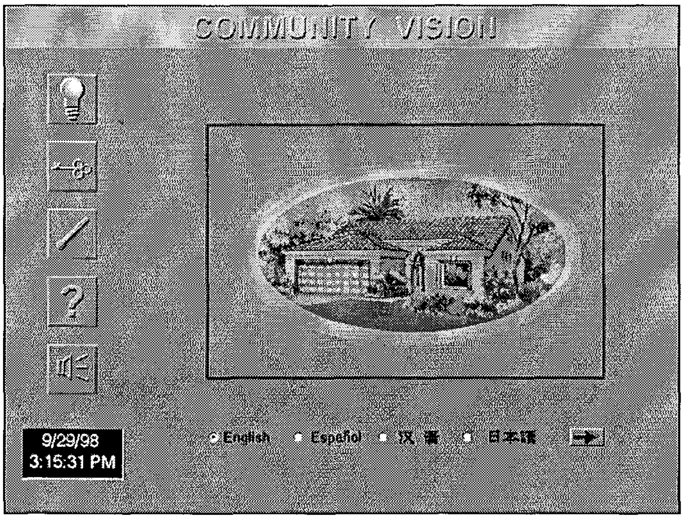

Figure 1

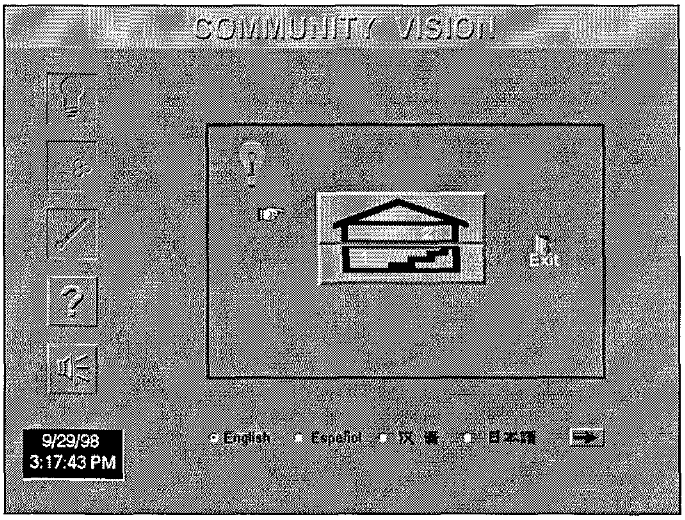

Figure 2

5. Characteristics of ETU interfaces

Even the simplest graphical user interface (GUI) requires some training. We are interested in studying the characteristics of interfaces that require no training or such a minimal amount of instruction that the program itself could offer this type of help when difficulty is detected. A great advance in interface design occurred about 20 years ago with the introduction of GUIs [7]. Widgets, that is pull-down or pop-up menus, icons, dialogue boxes, windows with re-sizing boxes, scroll-bars, and close boxes, came upon the scene. Window management still requires some training and is unsuitable for ETU interfaces in their current form, that is, with re-sizing boxes, scroll bars, etc. Similarly, the untrained user may not know how to use a menu or the effect it will have. Input devices for an ETU interface are primarily touch, pen and keypad. We believe voice input technology currently is not suitable for a broad range of diverse users because of accents and extraneous sounds that may interfere with recognition, as well as the imprecision of untrained users. In the future, voice may be an important input modality and, even now, specialized systems may be considered for use in environments where other input devices cannot be used. Other input devices may be used for more exotic applications, such as, space gloves, joysticks, 3-D pointing devices, camera recognition of desktop objects or hands, etc. Disabled users have limitations on the devices that can be used for input and output and may have a limited use of specific modalities.

5.1. Selection

Selection is the identification of one or more objects from a set of objects on a screen. If the contextual metaphor is a good one, the choice may be obvious. We found some types of selection difficult to present to the user. In some cases the user can pick one or more items from a set before designating the function. For example, in a home automation system, setting an operation by a timer can be time consuming. It is necessary to devise a method to indicate to the user that multiple lights can be selected before setting the timer operation. We have found no way to designate with visual symbols that the user can select more than one item. We attempted a mime gesture with a finger pointing to multiple items, but this only confused the subjects. In the case of selecting one item from a set, we found the use of a hand with a finger pointing to the objects (for touch screens) very effective as we did in Figure 2 with the choice of floors. Selection should always be acknowledged by feed-back such as a change of color, a box surrounding the selected object, or some such mechanism.

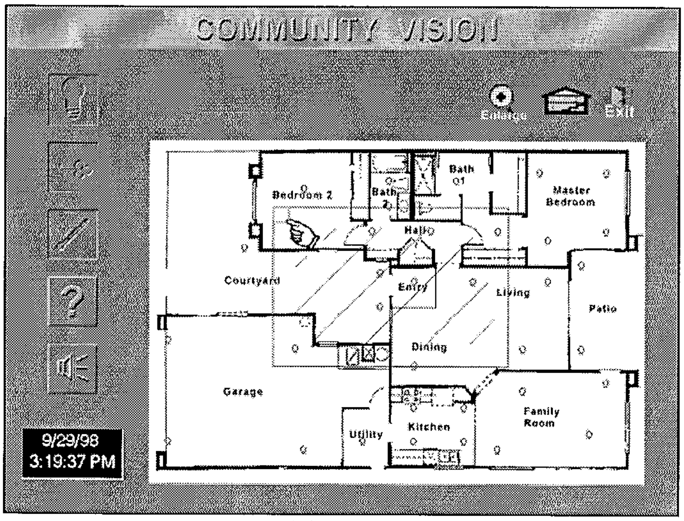

Figure 3

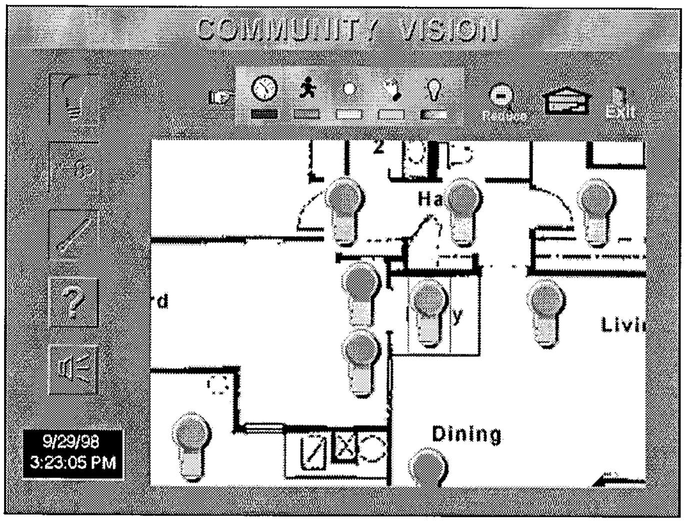

Figure 4

5.2. Navigation

What and how

Ease of navigation is the most critical issue in the design of ETU interfaces. The user must be able to perceive what sequences of operations can be performed without previous training. This may be accomplished with text descriptions as with an ATM. We believe visual symbols are more effective for our diverse group of users--we do supplement the interface with text and spoken help messages. The user is faced with the question of *what* to do next and *how* to do it. *What* is often called the task, while *how* is the function or operation provided by the interface for accomplishing the task. For example, the user may wish to enlarge an area on the layout of a house. The problem he or she now faces is how to do it. Figure 3 is the interface with a floor plan and a selection box indicates users should pick out the area of the house with lights they wish to be set. (The selection box is brilliant blue in contrast to any other color on the screen.) For implementation reasons, it is not possible to simply touch the area to be enlarged and have a satisfactory result. Once the user has dragged the selection box to the desired area, the area within the selection box must be enlarged. To enlarge the area, the user has to touch the enlarge icon (magnifying glass) in the upper right of the screen. Experienced computer users attempted to enlarge by touching the selection box itself, either in a corner (as with a resizing box) or in the middle. Inexperienced users were more likely to see the magnifier because it would not occur to them to touch the box to enlarge the area inside the box. We labeled the magnifier in bright yellow letters to catch the attention of users.

The major difficulty we experienced in the design of the interface is to lead users first to the *what* and then to the *how.*

1. *What:* Open the light function;
   How: Touch the light function;

2. *What:* Select a floor;
   How: Touch the floor to be viewed;

3. *What:* Select an area;
   How: Set the blue box over the entry light;

4. *What:* Magnify the area within the blue box;
   How: touch the magnifier;

5. *What:* Set the light to be on with daylight;
   What: Select the entry way light;
   How: Touch it
   What: Select the daylight function;
   How: Touch the icon for daylight;

6. *What:* Return to floor plan or exit;
   How: Touch the icon for reduce or return.

Example 1: Set the entry light to be turned on by daylight.

Animations

In our first design for a home automation interface we attempted to solve the problem of navigation by using animations of hands using mime gestures. That is, the user is informed of what and how to perform the next step by a mime. In step 2 above, a house icon appears with the number of floors that can be controlled by the system (See Figure 2). The user is expected to touch a floor that contains the lights to be set. A moving hand points to the floors to indicate to the user that he or she is to touch a floor. Several other animations were originally included: one was to demonstrate how the selection box was to be dragged, another was to show users they can select multiple lights before setting them. In all cases, when performing test, experienced subjects quickly deduced they touch a floor--some of them found the moving hand annoying. Inexperienced subjects found the moving hand useful. In all other cases, the moving icons were either misleading or distracting. For example, the subjects did not understand the purpose of the dragging animation and they touched the animation rather than dragging the box. The use of animated mime gestures was reduced to using only the pointing hand.

Selectable paths

Not all choices for navigation are backwards or forwards. Any possible choice should be shown on the screen; there should be some indication that this is a possible next choice. For example, web pages have a list of topics that the user can link to made clear by the convention of underlining words, however other visual symbols also provide links. An untrained user will be confused as to which visual objects can serve as a link. An ETU interface has icons or text that let the user jump to another location. However, it must be clear what can be touched with predictable results. The GUIs use the convention of graying the symbols or words in menus that cannot be used as selections, which is almost immediately understood. Our ETU interface also used this technique by graying icons whenever possible.

Figure 1 shows a list of languages. There are more languages than can displayed on the bottom of the page. Instead of scrolling with the use of a scroll bar, an icon with the word "more" is listed at the bottom of the language icons and when the user touches the word "more," new languages will replace the ones currently shown. The visual symbol for "more" is an arrow, but a better symbol needs to be found. The language list scrolls circularly.

5.3. Dragging

Once the user has a view of the floor plan, the area of interest must be enlarged. An alternative to this approach is to scroll an enlarged floor plan. We eliminated the use of a scroll bar because scrolling is not an obvious procedure to the untrained user. Also, it helps orient the user to see the entire floor before a section is enlarged. We introduced a (brilliant blue) box whose interior represents the area to be enlarged (See Figure 3). The problem we faced was to make it obvious to the user how to drag the blue box. There was a failed attempt to show a mime gesture to demonstrate dragging above the floor plan. Our next attempt was to put a hand with a finger pointing to the corner, that is, the dragging point for the box, but this didn't work well either. Experienced users tried dragging anywhere on the edge, while inexperienced users did not know what to do and even what to try. A small red box in the corner was then introduced together with the hand with the pointing finger and all of our subjects deduced the correct action immediately.

5.4. Contextual Metaphors

The notion of a *contextual metaphor* was developed in the Japan's Friend21 project [9]. Friend21 had the mission of creating interfaces for accessing information by the public, that is, anyone who had a television set. The Friend21 project goals were similar to those in our research. They predicted that large amounts of information would be accessible from one's home TV screen and interfaces were needed for this task. The solution was contextual metaphors--metaphors that come from experiences we have already had with the world around us, such as switches, radios, telephones, etc. One advantage of designing an interface for home automation is that the underlying metaphor of manipulating objects in a house is natural. Plaisant [8] examined varying designs for toggle buttons using three-dimensional graphic characteristics and sound--they looked and sounded like on/off switches. This is a contextual metaphor. ETU applications where contextual metaphors which directly pertain to the application are more likely to be successful than those where the application does not have such an obvious metaphor. In cases where there is no obvious metaphor, text-based information may be more successful.

How users will interpret an icon is one of the greatest challenges of ETU interfaces. As an example, when an enlarged view of lights is displayed (See Figure 4), the subject was asked to set a particular light, say, to dim the entry light. The subjects had the option of dimming the light by touching the bulb or by touching a dim icon on the upper part of the screen. The dim icon is also a bulb, but there is a bar beneath it with graduated shades of light and dark. They were not instructed how to accomplish this. They all touched the dim icon rather than attempting to dim by touching the light. We believe this is because they could see the dim icon, but a bulb does not indicate what action is to be taken next. It may also be argued that dimming a light is normally done by a slider or knob on the wall, not by touching a bulb. Generally, we found that icons for operations were more successful than asking the user to deduce which operations to use from displaying an object with no obvious functionality. Later we added a box around the operation icons at the top of the screen with a pointing finger to the box to indicate a next choice may be the selection of an operation.

5.5. Help

Although a good deal has been written about online help, we found that ETU interfaces posed special problems. Carroll [1] and Kearsley [4] have examined on-line help systems, but on a more sophisticated level than for ETU systems. Shneiderman [11] states that context-sensitive help has been found to be difficult for novice users. Roesler & McLellan [10] have a taxonomy of on-line help needs, which we found useful. The ETU interface help describes only the tasks that can be done next and how to do them. An ETU interface should not have to address the meaning of terms and other complex notions. There should be only one help icon to touch. This returns us to the basic ETU design of *what* and *how*. With each screen, help addresses the tasks that can be done and how they are done.

Screen 1: Figure 1
What: Open the light function;
How: Touch the light function;
Help response from this screen: Select one of the home automation functions of light, security, and climate control by touching the icon for that function.

Screen 2: Figure 2
What: Select a floor;
How: Touch the floor to be viewed;
Help response from this screen: Select the floor of your house that has the lights to be set by touching that floor. Or return to the previous page by touching the exit icon.

Screen 3: Figure 3
What: Select an area;
How: Set the blue box over the entry light;
Help response from this screen: Select the area you wish to control by dragging the blue box over the area. The blue box is dragged by touching the red box and moving your finger while in contact with the screen to the location you want. When the blue box is over the desired area, touch the enlarger. Or you can return to the previous page by touching the house icon.

Example 2: Response for help while navigating in order to set the entry light for motion.

One of the experiments we conducted was to run continuous help messages across the top of the screen where the words "Community Vision" now appear. Surprisingly, our test subjects ignored the text and preferred to experiment with different interactions unless they were unable to progress. Text help can be turned on or off by touching the question mark. We later added a voice output icon, shown as a loudspeaker. The user has a choice of four languages, both for voice and for text, which the user can select by touching the language (See Figure 1). Our help system is admittedly rudimentary; it has not been a focus of our research in this project. Clearly ETU interfaces should have good well-tested help systems, but we must leave this for another project.

6. Guidelines for ETU interfaces

The basic premise of an ETU interface is that the user with little or no computer experience can navigate through an application interface. We assume a highly diverse set of users, including those with disabilities, school-age children, and non-English speaking.

Our formative evaluations led to the following conclusions:

- Use as little text as possible, but do not replace text by visual symbols if the meaning is unclear. Use a good contextual metaphor whenever possible.

- Give the users a choice of language and use voice and text help, but allow users to turn off sound.

- The next move or moves must be obvious and there must be some indication on the interface what next possible moves are allowed.

- Operations may be better understood by operation icons rather than incorporating functions in an object icon with no apparent functionality associated with it. We could not solve the problem of how to show certain types of operations with visual symbols, but these can be broken down into a series of simpler operations.

- Direct manipulation may be confusing to an untrained user. We limited direct manipulation to dragging a box to another location; there should be a clear indication of how the dragging is to be done.

- Allow users to point to (touch) the interface or use some equally simple method for interacting with the interface. A cursor and mouse may be necessary if the ETU interface software is placed on a computer, which only uses this type of input for interaction. However, it is desirable not to use keyboards of the type now used with GUIs. If numbers are required input, a keypad can be used and text can be incorporated with soft keys or a pen

- Menus should be displayed as fixed part of a screen and not pop-up or pull-down wherever possible. A "more" selection can cycle through the items without a scroll bar.

7. Conclusions and future work

All the subjects we tested could use the interface easily after their initial session, that is, the second time through. This is not sufficiently good for us because our goal is to find what mechanisms allow the interface to be used the first time. Realistically, we believe this cannot be done perfectly without instructions in text. While the subjects were given formative evaluations, we realized that a number of options for improving the interface existed that we hadn't thought of. The subjects made many valuable suggestions, which we have tried to include these in our subsequent versions. Although the interface has the function of light control, security and climate control, many more options will be added. We hope to do much more testing on a still greater selection of users with and without computer experience.

We conclude with a speculation: as the computer becomes part of our culture, the design of ETU interfaces becomes a moving target. ETU and public access interfaces may be compared to using a public telephone. Even such a simple device can be baffling to a foreigner, who can't read instructions and is unfamiliar with the currency. As we pursue this research, we realize that there will always be barriers as to how far the dream of universal access can be realized because of cultural changes and evolving computer technology.

8. Acknowledgments

Meera M. Blattner is also with the Department of Biomathematics, M.D. Anderson Cancer Research Hospital, University of Texas Medical Center, Houston.

This work was performed with partial support of NSF Grant DUE-96-18688 and under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48.

9. References

[1] Carroll, J.M, *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill,* MIT Press, Boston, 1992

[2] Hix, Deborah, and Hartson, H. Rex, *Developing User Interfaces,* John Wiley, New York, 1993.

[3] Johnson, Jeff, "Simplifying the Controls of an Interactive Movie Game," *CHI 98, ACM Human Factors in Comp. Systems,* Los Angeles, CA, April 1998, pp 65-72.

[4] Kearsley, Greg. *On-Line Help Systems,* Ablex Publishing Company, Norwood, New Jersey, 1988.

[5] Kearsley, Greg. *Public Access Systems,* Ablex Publishing Company, Norwood, New Jersey, 1994

[6] Kimura, Takayuki Dan, "A Pen-Based Prosodic User Interface for Schoolchildren," *IEEE Multimedia,* Vol. 3, No. 4, 1996, pp 48-55.

[7] Miller, L.H., and Johnson, J., "The Xerox Star: and Influential User Interface Design," p 70-100, *Human-Computer Interface Design (*Rudiseill, Lewis, Polson, and McKay, eds.) Morgan Kaufman Publishers, Inc., San Francisco, 1995.

[8] Plaisant, Catherine, and Shneiderman, Ben, "Scheduling ON-OFF home control devices," *CHI 91, ACM Human Factors in Computing. Systems,* New Orleans, 1991, pp 459-460.

[9] Nonogaki, Hajime, and Ueda, Hirotada, "FRIEND21 Project: A Construction of the 21st Century Human Interface," *CHI 91, ACM Human Factors in Computing Systems,* New Orleans, LA, 1991, pp 407-413.

[10] Roesler, A.W., and McLellan, What Help Do Users Need?: Taxonomies for On-Line Information Needs & Access Methods, *CHI 95, ACM Human Factors in Computing Systems,* Denver, CO, 1995, pp 437-44.

[11] Shneiderman, Ben, *Designing the User Interface* (2nd Edition) Addison-Wesley, Reading, MA. 1992.